Discrete-Time System Properties

Professor Deepa Kundur

University of Toronto

Discrete-Time System Properties

Reference:

Section 2.2

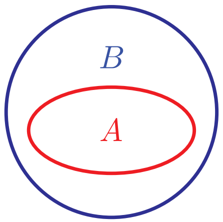
John G. Proakis and Dimitris G. Manolakis, *Digital Signal Processing: Principles, Algorithms, and Applications*, 4th edition, 2007.

Terminology: Implication

If "A" then "B" Shorthand: $A \implies B$

Example 1:
it is snowing $\implies$ it is at or below freezing temperature

Example 2:
$\alpha \geq 5.2 \implies \alpha$ is positive
Note: For both examples above, $B \not\Longrightarrow A$



Terminology: Equivalence

If "A" then "B" Shorthand: $A \implies B$

$\boxed{\text{and}}$

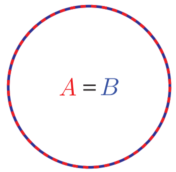
If "B" then "A" Shorthand: $B \implies A$

can be rewritten as

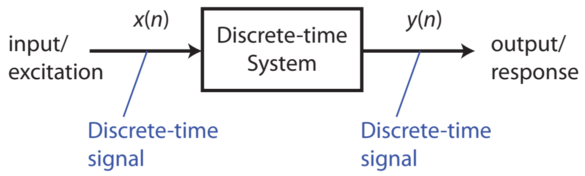
"A" if and only if "B" Shorthand: $A \iff B$

We can also say:

- A is EQUIVALENT to B
- $A = B$



Terminology: Input-Output Description



- Input-output description (exact structure of system is unknown or ignored):
$$y(n) = \mathcal{T}[x(n)]$$
- "black box" representation:
$$x(n) \xrightarrow{\mathcal{T}} y(n)$$

Classification of Discrete-Time Systems

Why is this so important?

- mathematical techniques developed to analyze systems are often contingent upon the general characteristics of the systems being considered
- For a system to possess a given property, the property must hold for every possible input to the system.
  - to disprove a property, need a single counter-example
  - to prove a property, need to prove for the general case

Classification of Discrete-Time Systems

Common System Properties:

static	vs.	dynamic
time-invariant	vs.	time-variant
linear	vs.	nonlinear
causal	vs.	non-causal
stable	vs.	unstable systems
⋮		⋮

Static vs. Dynamic

- Static system (a.k.a. memoryless): the output at time n depends only on the input sample at time n; otherwise the system is said to be dynamic
- a system is static iff (if and only if)
$$y(n) = \mathcal{T}[x(n), n]$$
for every time instant n.

Static vs. Dynamic

- Consider the general system:

$$y(n) = \mathcal{T}[x(n-N), x(n-N+1), \cdots, x(n-1), x(n), x(n+1), \cdots, x(n+M-1), x(n+M)], \qquad N, M > 0$$

  - For $N = M = 0$, $y(n) = \mathcal{T}[x(n)]$, the system is static.
  - For $0 < N, M < \infty$, the system is said to be dynamic with finite memory of duration $N + M + 1$.
  - For either N and/or M equal to infinite, the system is said to have infinite memory.

Static vs. Dynamic

Examples: memoryless or not?

- $y(n) = A\, x(n)$, $A \neq 0$
- $y(n) = A\, x(n) + B$, $A, B, \neq 0$
- $y(n) = x(n)\cos(\frac{\pi}{25}(n-5))$
- $y(n) = x(-n)$
- $y(n) = x(n+1)$
- $y(n) = \frac{1}{1-x(n+2)}$
- $y(n) = e^{3x(n)}$
- $y(n) = \sum_{k=-\infty}^{n} x(k)$

Ans: Y, Y, Y, N, N, N, Y, N

Time-invariant vs. Time-variant Systems

- Time-invariant system: input-output characteristics do not change with time

- a system is time-invariant iff

$$x(n) \xrightarrow{\mathcal{T}} y(n) \implies x(n-k) \xrightarrow{\mathcal{T}} y(n-k)$$

for every input $x(n)$ and every time shift k.

Time-invariant vs. Time-variant Systems

Examples: time-invariant or not?

- $y(n) = A\, x(n)$, $A \neq 0$
- $y(n) = A\, x(n) + B$, $A, B, \neq 0$
- $y(n) = x(n)\cos(\frac{\pi}{25}n)$
- $y(n) = x(-n)$
- $y(n) = x(n+1)$
- $y(n) = \frac{1}{1-x(n+2)}$
- $y(n) = e^{3x(n)}$
- $y(n) = \sum_{k=-\infty}^{n} x(k)$

Ans: Y, Y, N, N, Y, Y, Y, Y

Linear vs. Nonlinear Systems

- Linear system: obeys superposition principle
- a system is linear iff

$$\mathcal{T}[a_1\ x_1(n) + a_2\ x_2(n)] = a_1\ \mathcal{T}[x_1(n)] + a_2\ \mathcal{T}[x_2(n)]$$

for <u>any</u> arbitrary input sequences $x_1(n)$ and $x_2(n)$, and <u>any</u> arbitrary constants a_1 and a_2

Linear Systems: Homogeneity

A system is linear iff

$$\mathcal{T}[a_1\ x_1(n) + a_2\ x_2(n)] = a_1\ \mathcal{T}[x_1(n)] + a_2\ \mathcal{T}[x_2(n)]$$

- Homogeneity: Let $a_2 = 0$.

$$\mathcal{T}[a_1\ x_1(n)] = a_1\ \mathcal{T}[x_1(n)]$$

$$x(n) \xrightarrow{\mathcal{T}} y(n) \implies a_1 x(n) \xrightarrow{\mathcal{T}} a_1 y(n)$$

for <u>any</u> constant a_1.

Linear Systems: Additivity

A system is linear iff

$$\mathcal{T}[a_1\ x_1(n) + a_2\ x_2(n)] = a_1\ \mathcal{T}[x_1(n)] + a_2\ \mathcal{T}[x_2(n)]$$

- Additivity: Let $a_1 = a_2 = 1$.

$$\mathcal{T}[x_1(n) + x_2(n)] = \mathcal{T}[x_1(n)] + \mathcal{T}[x_2(n)]$$

$$\begin{matrix} x_1(n) \xrightarrow{\mathcal{T}} y_1(n) \\ x_2(n) \xrightarrow{\mathcal{T}} y_2(n) \end{matrix} \implies x_1(n) + x_2(n) \xrightarrow{\mathcal{T}} y_1(n) + y_2(n)$$

for <u>any</u> input sequences $x_1(n)$ and $x_2(n)$.

Linear Systems: Additivity

Therefore:

Linearity = Homogeneity + Additivity

Need both!
If a system is <u>not</u> homogeneous, it is <u>not</u> linear.
If a system is <u>not</u> additive, it is <u>not</u> linear.

Linear vs. Nonlinear Systems

Examples: linear or not?

- $y(n) = A\,x(n),\ A \neq 0$
- $y(n) = A\,x(n) + B,\ A, B, \neq 0$
- $y(n) = x(n)\cos(\frac{\pi}{25}n)$
- $y(n) = x(-n)$
- $y(n) = x(n+1)$
- $y(n) = \frac{1}{1-x(n+2)}$
- $y(n) = e^{3x(n)}$
- $y(n) = \sum_{k=-\infty}^{n} x(k)$

Ans: Y, N, Y, Y, Y, N, N, Y

Causal vs. Noncausal Systems

- Causal system: output of system at any time n depends only on present and past inputs
- a system is causal iff

$$y(n) = \mathcal{T}\left[x(n), x(n-1), x(n-2), \ldots\right]$$

for all n

Causal vs. Noncausal Systems

Examples: causal or not?

- $y(n) = A\,x(n),\ A \neq 0$
- $y(n) = A\,x(n) + B,\ A, B, \neq 0$
- $y(n) = x(n)\cos(\frac{\pi}{25}(n+1))$
- $y(n) = x(-n)$
- $y(n) = x(n+1)$
- $y(n) = \frac{1}{1-x(n+2)}$
- $y(n) = e^{3x(n)}$
- $y(n) = \sum_{k=-\infty}^{n} x(k)$

Ans: Y, Y, Y, N, N, N, Y, Y

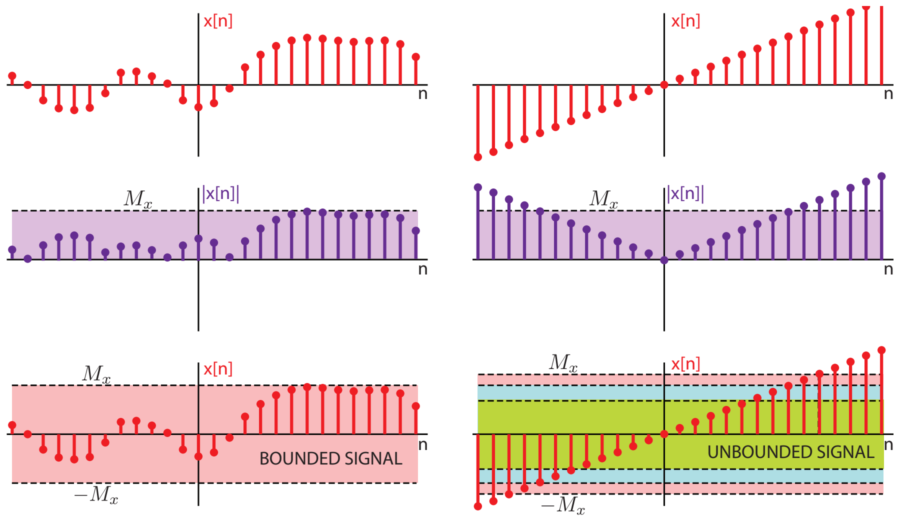
Stable vs. Unstable Systems

- Bounded Input-Bounded output (BIBO) Stable: every bounded input produces a bounded output
- a system is BIBO stable iff

$$|x(n)| \leq M_x < \infty \implies |y(n)| \leq M_y < \infty$$

for all n.

Discrete-Time Bounded Signals



Stable vs. Unstable Systems

Examples: stable or not?

- $y(n) = A\,x(n),\ A \neq 0$
- $y(n) = A\,x(n) + B,\ A, B, \neq 0$
- $y(n) = x(n)\cos(\frac{\pi}{25}n)$
- $y(n) = x(-n)$
- $y(n) = x(n+1)$
- $y(n) = \frac{1}{1-x(n+2)}$
- $y(n) = e^{3x(n)}$
- $y(n) = \sum_{k=-\infty}^{n} x(k)$

Ans: Y, Y, Y, Y, Y, N, Y, N

Final Remarks

- For a system to possess a given property, the property must hold for every possible input and parameter of the system.
  - to disprove a property, need a single counter-example
  - to prove a property, need to prove for the general case